



TotalView 8.7 Update

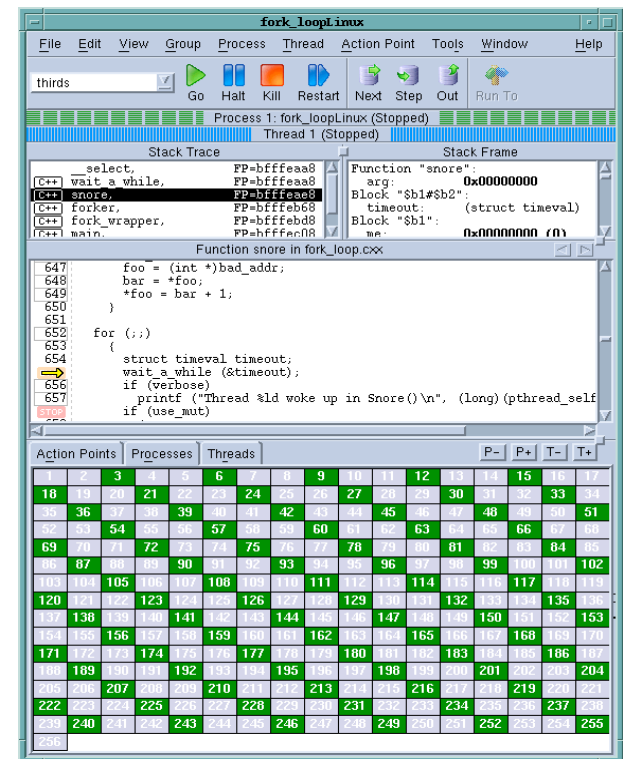
INCITE Call
Oct 29, 2009

Chris Gottbrath
Director, Product Management



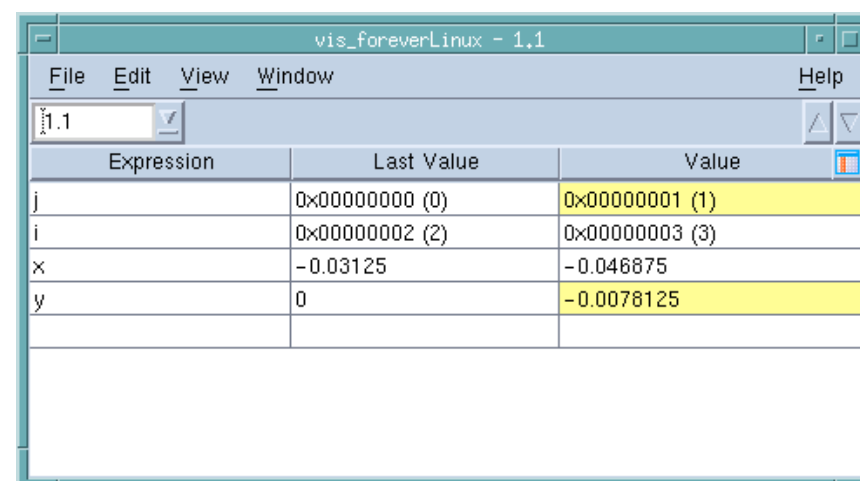
TotalView debugger

- **Use C, C++, Fortran 77, Fortran90, UPC**
 - Advanced language features
- **Wide compiler and platform support**
 - BlueGene L and P
 - Linux x86, x86-64, Power, Cell
 - AIX, Solaris, HP-UX, IRIX, Tru64
 - Cray XT, Apple OS X, and others
- **Parallel debugging**
 - MPI, pthreads, OpenMP, UPC
- **Memory debugging capabilities**
 - Integrated into the debugger
- **Graphical User Interface**
 - Simple things are easy
 - Advanced operations are available
 - Visualization
- **Remote Display Client**
 - Debug graphically from anywhere
- **Scripting**
 - Simply things with TVScript
 - Fully programmable CLI



Productivity Features in TotalView

- **Highlighting variables that change**
 - Expression list
 - Data Window
- **Setting breakpoints in dynamic libraries**
 - Even before they are loaded
- **“View Across”**
 - a single operation from the source pane
- **Setting Watchpoints**
 - a single operation from the source pane
- **Breakpoints on all methods of a class**
 - a single operation
- **STL View**



vis_foreverLinux - 1.1

Expression	Last Value	Value
j	0x00000000 (0)	0x00000001 (1)
i	0x00000002 (2)	0x00000003 (3)
x	-0.03125	-0.046875
y	0	-0.0078125

- Dive
- Add to Expression List
- Across Processes
- Across Threads
- Set Breakpoint
- Set Barrier
- Create Watchpoint
- Enable
- Disable
- Delete
- Properties

Productivity Features in TotalView

- **Indirect MPI Launch**

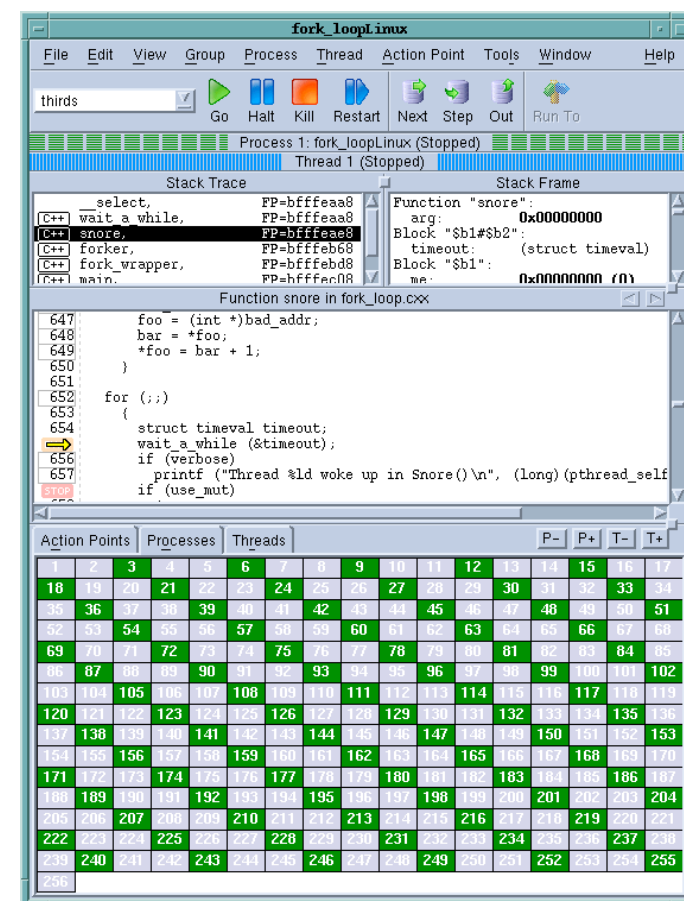
- One way to launch program within TotalView for all MPIs
- Reduces reliance on process acquisition interface -- fewer start up troubles
- Always start with the users code (no longer focus on mpirun first)

- **Visual display of MPI processes**

- Compact
- Represents process state and selection and group membership

- **Create process groups from Call graph**

- **Cycle Detection in MPI Message Queue Graph**



Batch Debugging with TVScript

- **User extensible script to drive a target program to completion under the TotalView debugger.**
 - No user interaction
 - Generates output in a text file
- **Handles all the event management overhead so the user doesn't have to.**
 - User defines breakpoints
 - User defines actions
- **Allows Users to**
 - Gather debugging data in batch environments
 - Do structured problem analysis
 - Localize problems for interactive debugging

tvscript - example

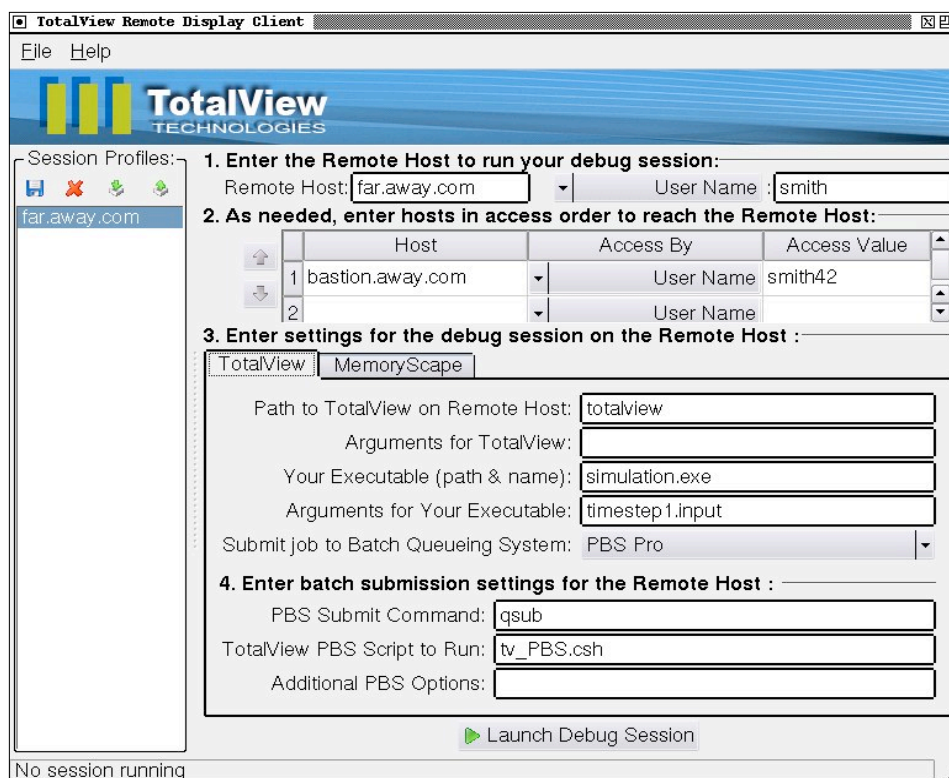
- The following command flag tells tvscript to report the contents of the *foreign_addr* structure each time the program gets to line 85
-create_actionpoint "#85=>print foreign_addr"
- Typical output blocks sample with tvscript:

```
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
! Print
!
! Process:
!   ./server (Debugger Process ID:  1, System ID:  12110)
! Thread:
!   Debugger ID:  1.1, System ID:  3083946656
! Time Stamp:
!   06-26-2008 14:04:09
! Triggered from event:
!   actionpoint
! Results:
!   foreign_addr = {
!     sin_family = 0x0002 (2)
!     sin_port = 0x1fb6 (8118)
!     sin_addr = {
!       s_addr = 0x6658a8c0 (1717086400)
!     }
!     sin_zero = ""
!   }
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
```

Remote Display Scenarios

- **When you are on-site with the HPC resource**
 - When you want to use a cluster at another site
 - If your site-wide network is sufficiently complex that it isn't easy to set up a remote connection
 - When you are working with colleagues who are remote .. you may want to have them use the Remote Display Client
- **When you are a off-site HPC user**
 - Any time you want to debug on the HPC system (unless you want to do a non-interactive batch session like we talked about earlier)

Long Distance Remote Display



The screenshot shows the TotalView Remote Display Client window. It has a menu bar with 'File' and 'Help'. Below the menu bar is a toolbar with icons for session profiles. The main area is titled 'Session Profiles:' and contains a list of profiles, with 'far.away.com' selected. To the right of the list are four numbered steps for configuring a remote session:

- 1. Enter the Remote Host to run your debug session:** Remote Host: User Name:
- 2. As needed, enter hosts in access order to reach the Remote Host:** A table with columns 'Host', 'Access By', and 'Access Value'. It contains two rows:

	Host	Access By	Access Value
1	bastion.away.com	User Name	smith42
2		User Name	
- 3. Enter settings for the debug session on the Remote Host :** Tabs for 'TotalView' and 'MemoryScape'. Fields include:
 - Path to TotalView on Remote Host:
 - Arguments for TotalView:
 - Your Executable (path & name):
 - Arguments for Your Executable:
 - Submit job to Batch Queueing System:
- 4. Enter batch submission settings for the Remote Host :**
 - PBS Submit Command:
 - TotalView PBS Script to Run:
 - Additional PBS Options:

At the bottom right is a 'Launch Debug Session' button. The status bar at the bottom left says 'No session running'.

- **The Remote Display Client**
 - Included in TV distribution
 - Also available on the web
- **Sets up a graphical connection**
 - Via ssh
 - Through one or more hosts
 - To a remote machine
- **Provides for a connection that is**
 - Easy
 - Fast
 - Secure
- **The Remote Display Client is available for:**
 - Linux x86
 - Linux x86-64
 - Windows XP
 - Windows Vista
- **Does job submission with batch Environments**
 - PBS Pro
 - LoadLeveler

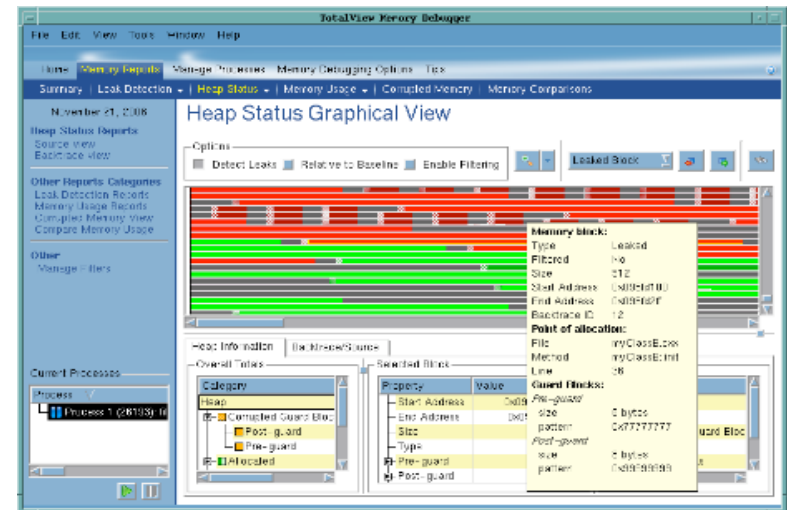


TotalView 8.7

- **Memory Debugging**
 - Interoperability between TotalView and MemoryScape
 - Improved memory hoarding support
- **Other Improvements**
 - Compilers
 - XLF 12.1
 - XLC 10.1
 - GCC 4.4
 - Support for generic linux PPC32 in TotalView
 - Support for Heterogeneous debugging in TotalView
 - Support for IP only networks
 - Command line subset attach

MemoryScape

- **MemoryScape is**
 - Streamlined
 - Lightweight
 - Intuitive
 - Collaborative
 - Memory Debugging
- **MemoryScape features**
 - It shows
 - Memory errors
 - Memory status
 - Memory leaks
 - Buffer overflows
 - Guard Blocks
 - Red Zones (new)
 - MPI memory debugging
 - Remote memory debugging



- Tech
 - Low overhead
 - No Instrumentation
- Interface
 - Inductive
 - Collaboration
 - Multi-process